|  |
| --- |
| 哈尔滨工业大学计算机科学与技术学院暨软件学院 |
| AIDA详细设计说明 |
|  |
|  |

2019年06月10日-2019年07月05日

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 角色 | 学号 | 姓名 | 组内分数 | 小组成绩 |
| 组长 | 171110315 | 彭伟智 |  |  |
| 组员 | 171110314 | 毛煜洲 |  |
| 组员 | 171110311 | 李宝航 |  |
| 组员 | 171110301 | 卜君健 |  |
| 组员 | 171110331 | 张婧 |  |
| 组员 |  |  |  |
| 组员 |  |  |  |

|  |  |
| --- | --- |
| 指导教师 | 韩希先 |

|  |  |
| --- | --- |
| 教师评语 |  |

目录

[1.总述 3](#_Toc11915439)

[1.1目标 3](#_Toc11915440)

[1.2背景 3](#_Toc11915441)

[2.程序系统组织结构 3](#_Toc11915442)

[3.程序模块设计 3](#_Toc11915443)

[3.1客户端 3](#_Toc11915444)

[3.2服务器端 7](#_Toc11915445)

[4.数据模型设计 8](#_Toc11915446)

[5.界面实现说明 10](#_Toc11915447)

1.总述

1999年腾讯发布了一款自主开发的仿ICQ的基于Internet的即时通信网络工具OICQ，即QQ的前身。在当时互联网浪潮兴起，除了OICQ，国内还涌现出一大波仿icq的即时通讯程序, 如Picq、Ricq、Ticq(TQ) 、Qicq、Micq、PCicq、Oicq、OMMO等。经过时间的考验，QQ在如此众多的在线即时通讯软件中脱颖而出，最终淘汰了以上种种其他对手一直活到了现在,占领了中国在线即时通讯软件市场几乎全部的市场，不仅靠运气 ，也有实力。

1.1目标

体验敏捷Scrum开发，并且运用本学期学习的计算机网络，与操作系统的相关技术，实现类似QQ软件的实现，以及对本学期知识的运用。

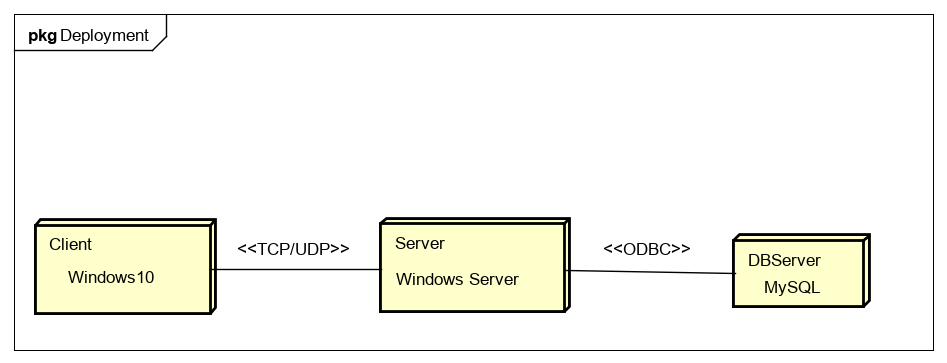
1.2背景

如今随着QQ占领整个市场，QQ本身集成了越来越多的功能如邮箱、空间、网盘等等，已经上升成为社交、工具软件，单纯作为聊天软件使用时很多功能冗余，软件本身也变得越来越庞大，并且运行时占用大部分资源在一些旧机器上严重影响性能。因此，我们决定设计开发一款简单实用、在即时聊天方面能代替qq的自用软件。

同时，由于校园网学校的带宽紧缩，下载外网数据变得越来越慢。在外网速度越来越慢的时候我们想到了内网传输。当校内已经有同学有资源时，可以通过内网传输，这时候速度会不受外网的限制，达到100Mbps的速度，利用校园局域网，这也会极大的方便同学们使用，减小学校的外网负担。

在21世纪，独生子女多了起来，他们大多没有兄弟姐妹可以交流，我们设计一个聊天机器人，调用图灵机器人API实现简单的人机会话。

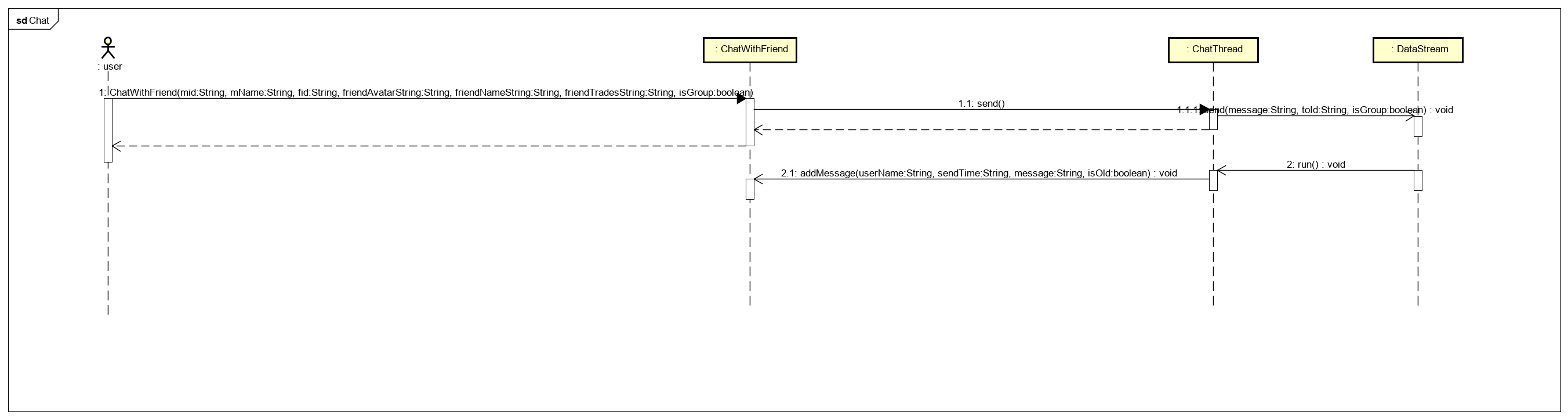
2.程序系统组织结构



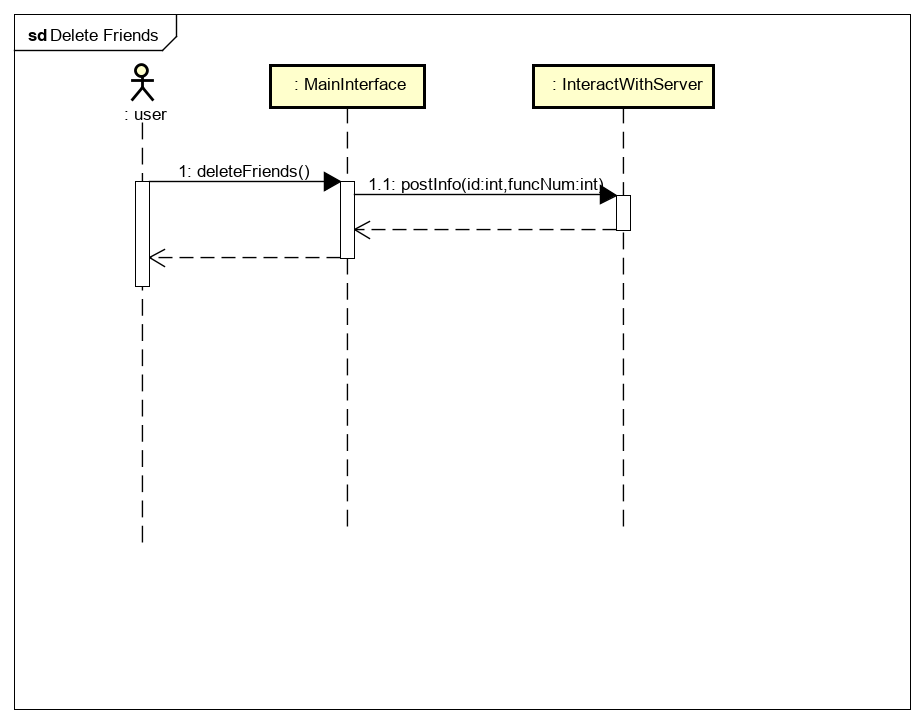
3.程序模块设计

3.1客户端

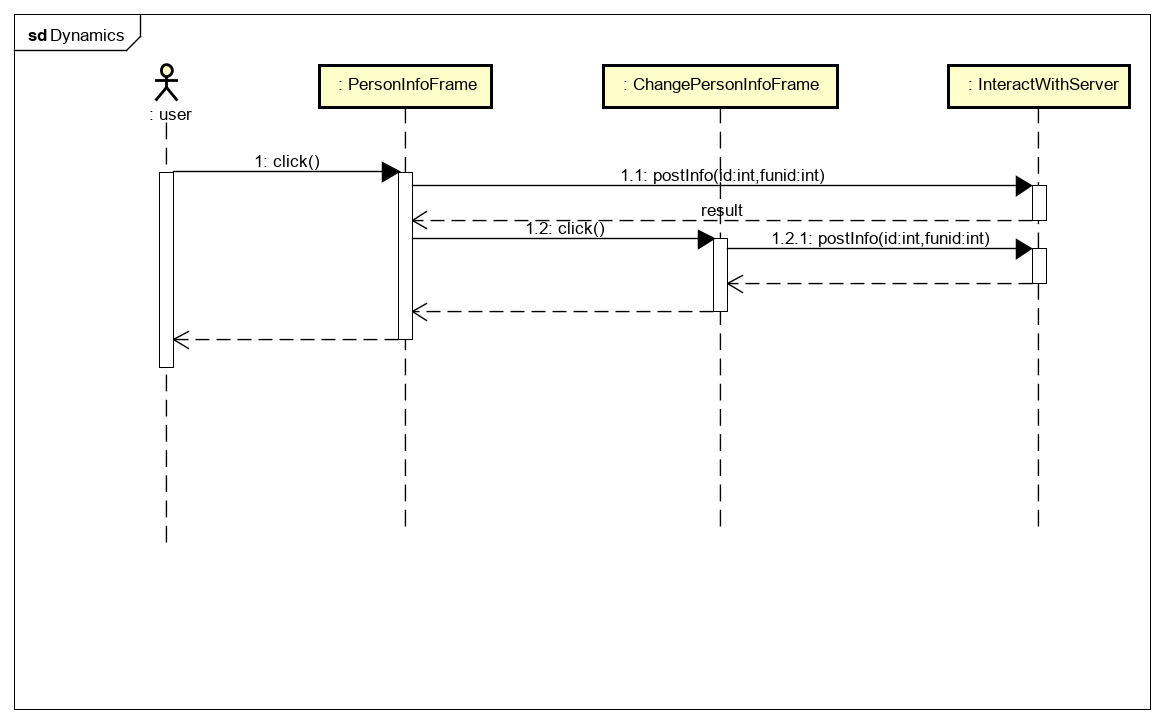
聊天：



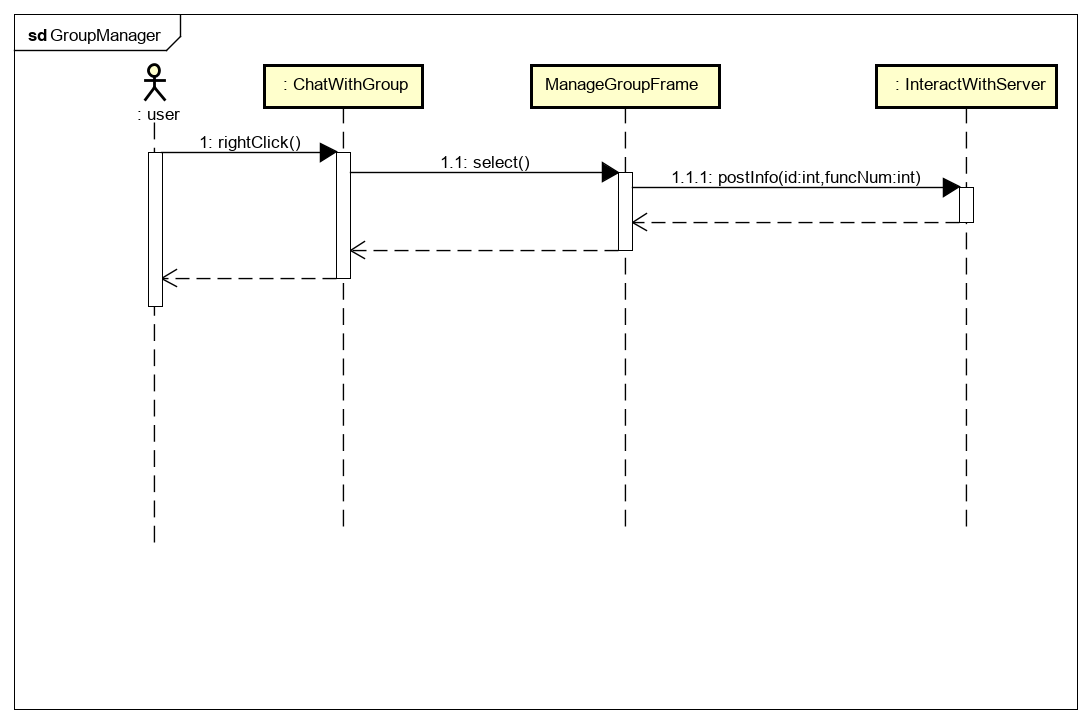
删除朋友：



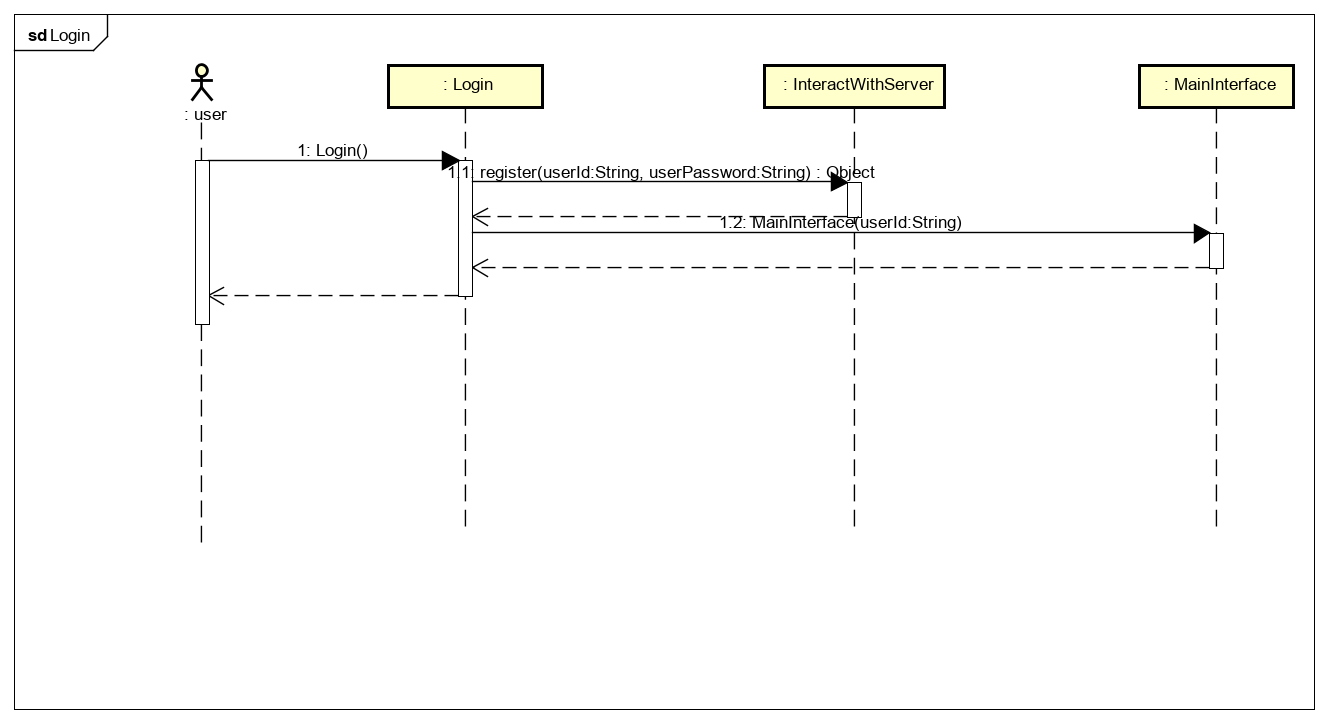
动态：



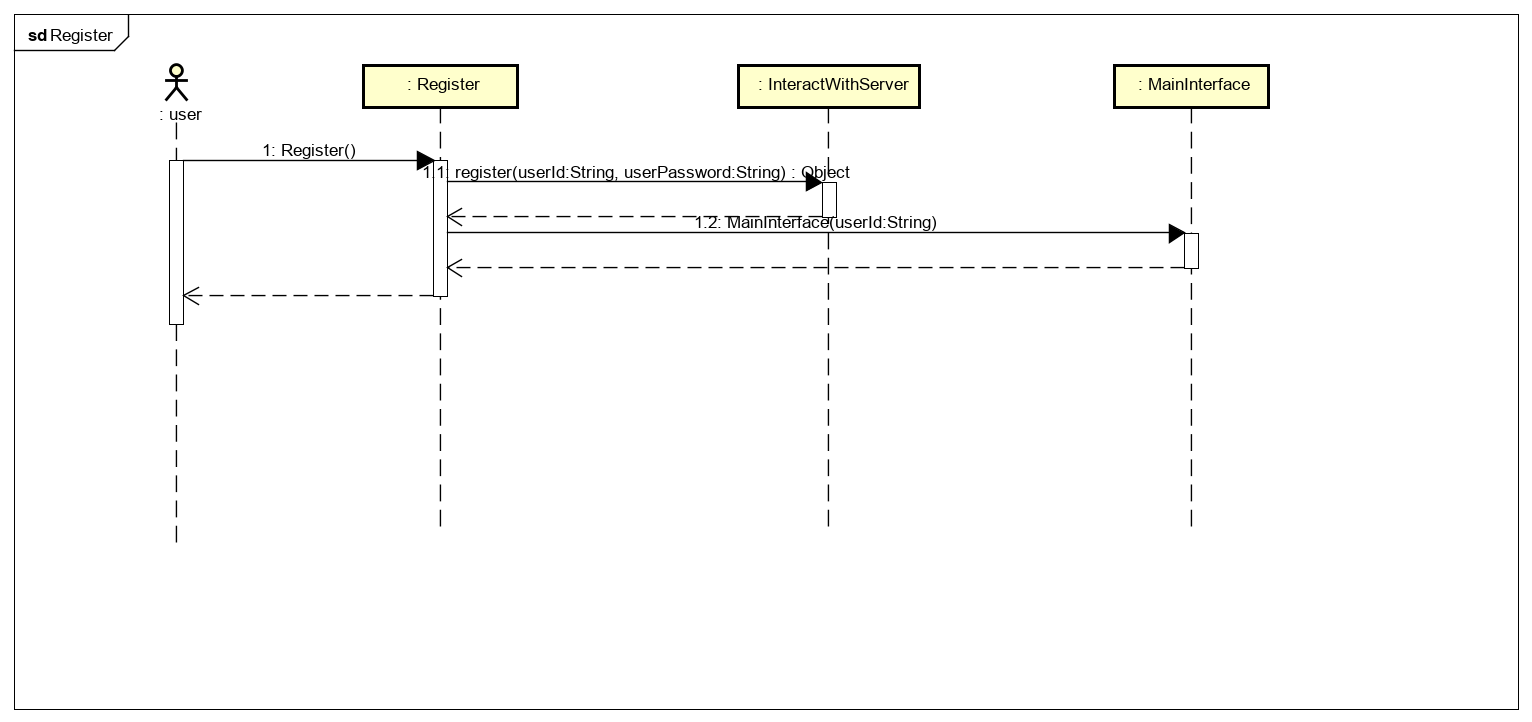
群管理：



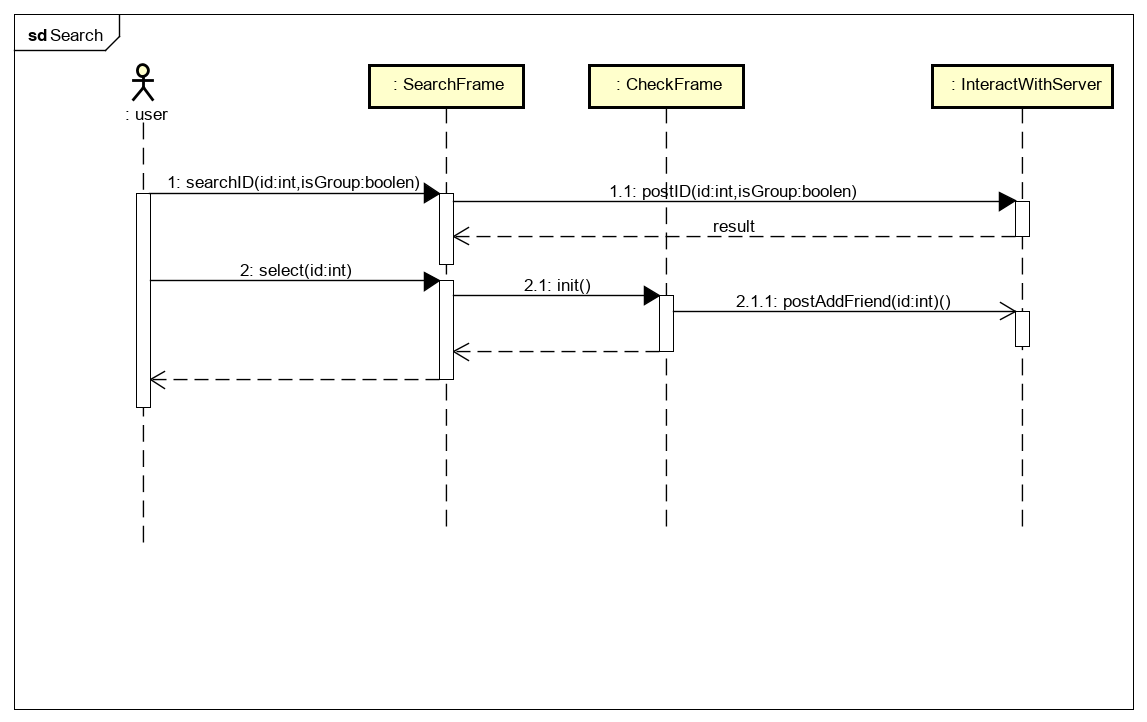
登录：



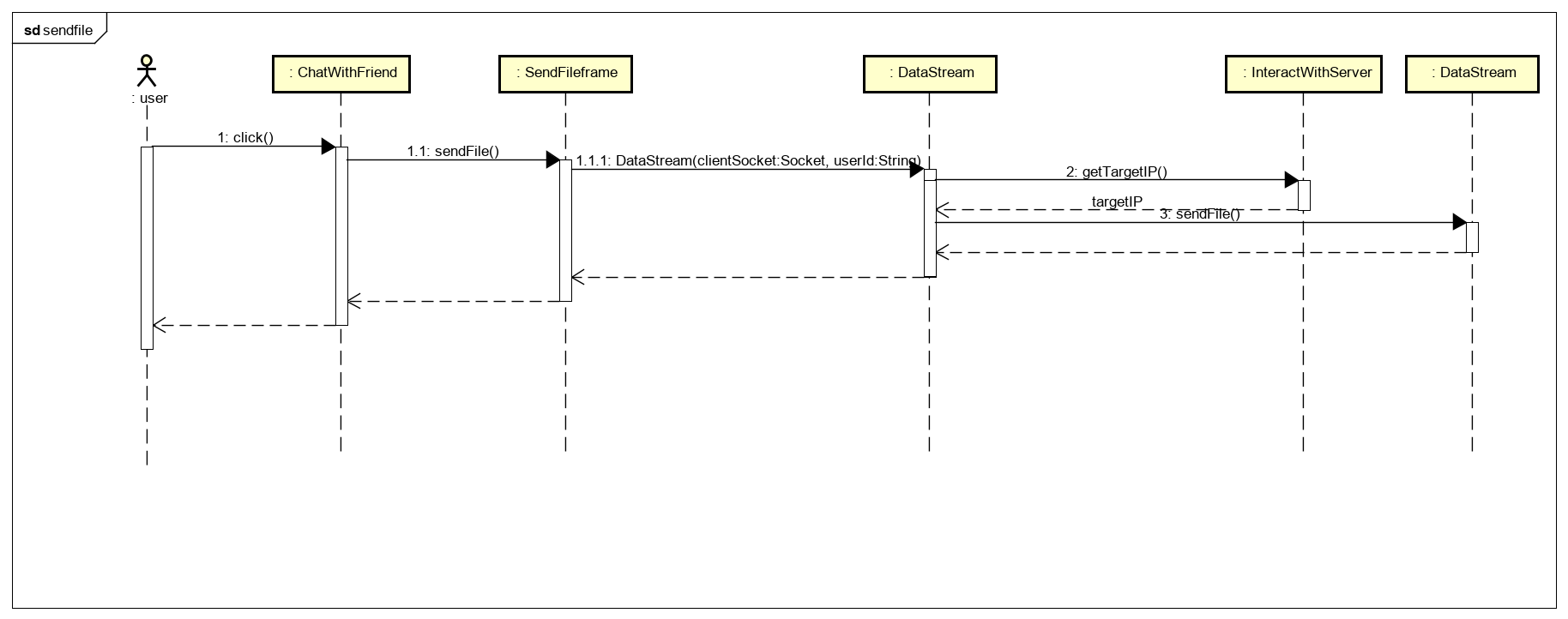
注册：



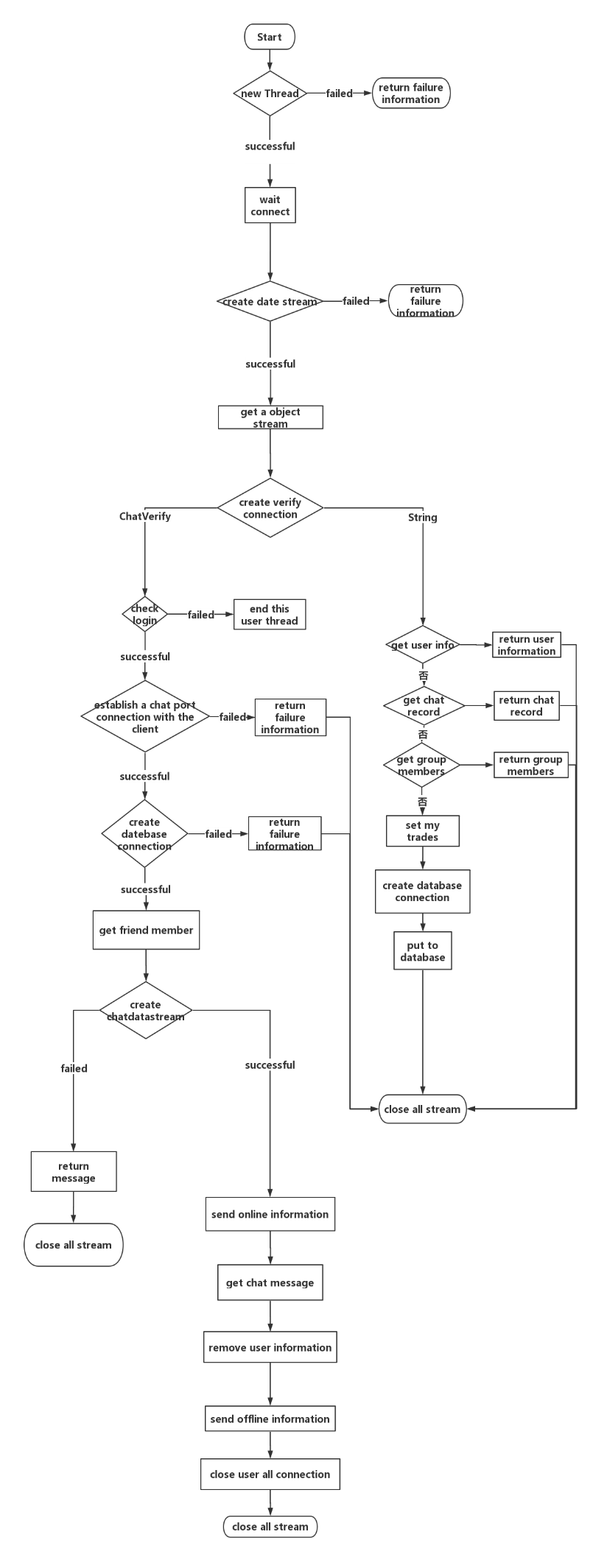
搜索好友：



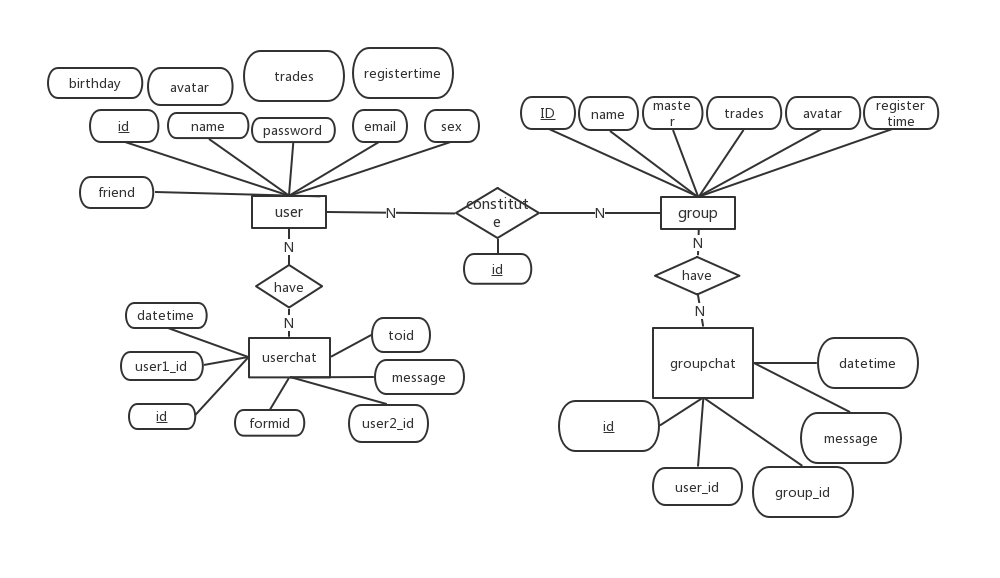
传输文件：



3.2服务器端



4.数据模型设计



数据设计E-R图

表1 用户表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***表名*** | ***字段名*** | ***含义*** | ***数据类型*** | ***大小*** | ***是否为空*** | ***备注*** |
| *user* | *user\_id* | *用户id* | *int* | *4* | *NO* | *Primary key* |
|  | *user\_name* | *用户名* | *Varchar(20)* | *20* | *NO* |  |
|  | *user\_password* | *用户密码* | *varchar(32)* | *32* | *NO* |  |
|  | *user\_email* | *用户邮箱* | *varchar(30)* | *30* | *NO* |  |
|  | *user\_sex* | *用户性别* | *char(1)* | *1* | *NO* |  |
|  | *user\_birthday* | *用户出生日期* | *varchar(12)* | *12* | *NO* |  |
|  | *user\_avatar* | *用户替身* | *text* |  | *NO* |  |
|  | *user\_trades* | *个性签名* | *text* |  | *NO* |  |
|  | *user\_registertime* | *用户注册时间* | *datetime* |  | *NO* |  |

用户表的详细设计如图上所示，用户id为该关系表的主键，为大小为4的int类型数据，该表中包括所有用户信息。

表2 群表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***表名*** | ***字段名*** | ***含义*** | ***数据类型*** | ***大小*** | ***是否为空*** | ***备注*** |
| *group* | *group\_id* | *群id* | *int* | *10* | *NO* | *Primary key* |
|  | *group\_name* | *群名* | *Varchar(20)* | *20* | *NO* |  |
|  | *group\_master* | *群主* | *int(10)* | *10* | *NO* | *foreign key* |
|  | *group\_trades* | *群个性签名* | *text* |  | *NO* |  |
|  | *group\_avatar* | *群替身* | *text* |  | *NO* |  |
|  | *group\_registertime* | *群注册时间* | *datetime* |  | *NO* |  |

群表的详细设计如图上所示，群的id为该表的主键，为大小为10的int类型数据，该表包括一个群实体的所有属性。

表3 好友关系表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***表名*** | ***字段名*** | ***含义*** | ***数据类型*** | ***大小*** | ***是否为空*** | ***备注*** |
| *useruser* | *myself* | *本人id* | *int* | *10* | *NO* | *Primary key*  *foreign key* |
|  | *myfriend* | *好友id* | *int* | *10* | *NO* | *Primary key*  *foreign key* |

好友关系表的详细设计如图上所示，myself和myfriend字段一起构成该表的主键，同时myself和myfriend又分别作为外键连接两个用户表示他们之间的好友关系。

表4 用户群聊表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***表名*** | ***字段名*** | ***含义*** | ***数据类型*** | ***大小*** | ***是否为空*** | ***备注*** |
| *usergroup* | *user\_id* | *用户id* | *int* | *4* | *NO* | *Primary key*  *foreign key* |
|  | *group\_id* | *群id* | *Varchar(20)* | *20* | *NO* | *Primary key*  *foreign key* |

用户群聊表的详细设计如图上所示，用户id和群id一起构成该表的主键，同时又作为外键连接一个用户和一个群表示一个用户属于一个群中。

表5 用户聊天表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***表名*** | ***字段名*** | ***含义*** | ***数据类型*** | ***大小*** | ***是否为空*** | ***备注*** |
| *userchat* | *uchat\_id* | *用户聊天id* | *int* | *4* | *NO* | *Primary key* |
|  | *uchat\_fromid* | *发送方用户id* | *int(10)* | *10* | *NO* | *foreign key* |
|  | *uchat\_toid* | *接收方用户id* | *int(10)* | *10* | *NO* | *foreign key* |
|  | *uchat\_message* | *消息记录* | *text* | *10* | *NO* |  |
|  | *uchat\_datetime* | *消息时间* | *datetime* |  | *NO* |  |

用户聊天表的详细设计如图上所示，用户聊天id作为该表的主键，为大小是4的int类型数据，外键为发送方用户id和接收方用户的id。该表记录了用户聊天的消息记录和消息时间。

表6 群聊天表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***表名*** | ***字段名*** | ***含义*** | ***数据类型*** | ***大小*** | ***是否为空*** | ***备注*** |
| *groupchat* | *gchat\_id* | *群聊id* | *int* | *4* | *NO* | *Primary key* |
|  | *gchat\_uid* | *群聊的用户id* | *int(10)* | *10* | *NO* | *foreign key* |
|  | *gchat\_gid* | *群聊的群id* | *int(10)* | *10* | *NO* | *foreign key* |
|  | *gchat\_message* | *消息记录* | *text* | *10* | *NO* |  |
|  | *gchat\_datetime* | *消息时间* | *datetime* |  | *NO* |  |

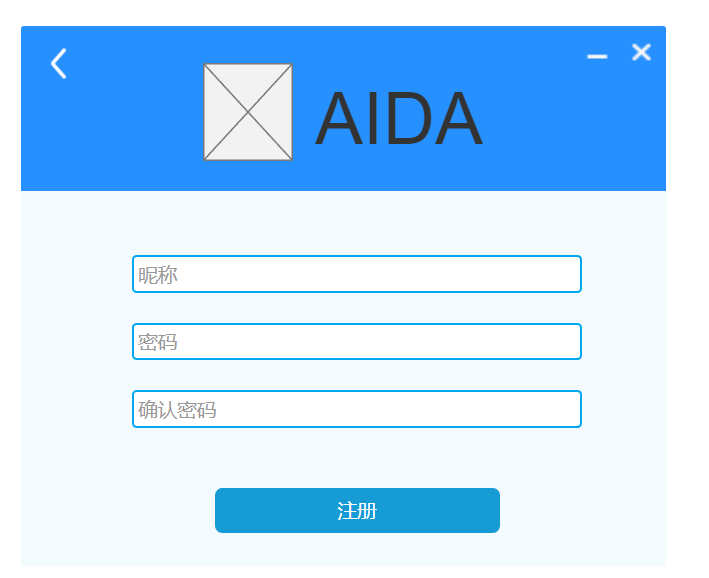
群聊天表的详细设计如图上所示，群聊id作为该表的主键，为大小是4的int类型数据，外键为群聊的用户id和群聊的群id。该表记录了群聊的消息记录和消息发送的时间。

5.界面实现说明

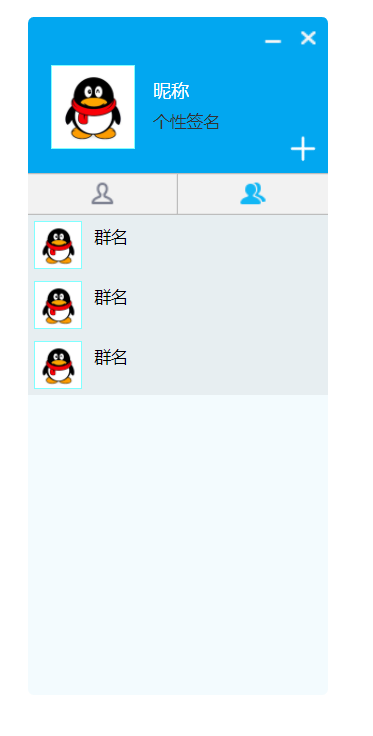
登录界面：



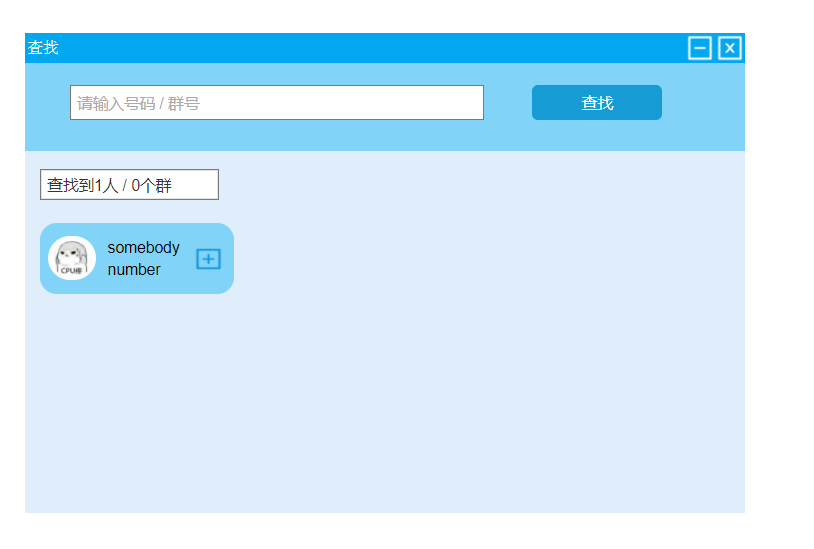
登录界面：



主页1： 主页2：

查找群聊/好友：



添加界面：



个人信息1： 个人信息2：



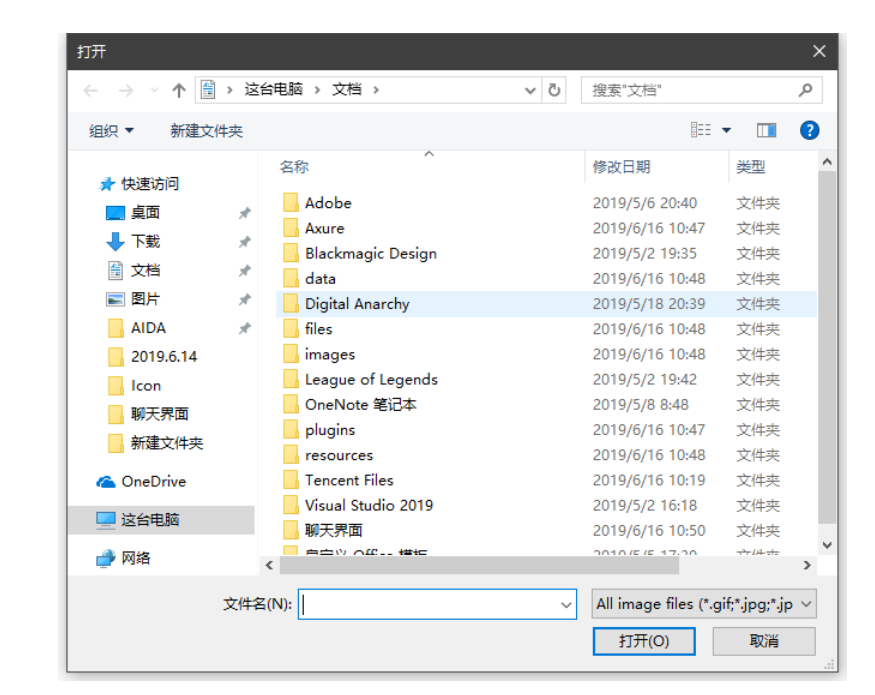
编辑资料：



语音界面：



发送文件界面：



语音对话界面：



视频聊天界面:

